AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method comprising:

providing a partition on an Integrated Device Electronics ("IDE") storage device of a computer system, wherein said partition is invisible to an operating system of the computer system unless the partition is unlocked;

providing a software task having knowledge about ~~a proper~~ an unlock handshake between the software task and an IDE controller, ~~to unlock the partition such that the partition that was previously invisible to the operating system becomes visible to the operating system;~~

~~establishing~~ performing ~~a proper~~ the unlock handshake including an alteration of an electrical signal on an IDE controller interface line; and

unlocking the partition in response to an unlock request received from the software task after the software task performs the unlock handshake to unlock the partition, wherein the partition is visible to the operating system when unlocked.

2. (Previously Presented) The method of claim 1, wherein the storage device is a hard disk drive having an IDE disk controller.

3. (Canceled)

4. (Currently Amended) The method of claim 1, wherein the software task requests a master token from the IDE controller when the computer system is first turned on and the unlock handshake between the software task and the IDE controller ~~is established by~~ further comprises:

passing the master token back to the IDE controller as a parameter.

5. (Currently Amended) The method of claim 2, wherein the software task requests a master token from the disk controller when the computer system is first turned on, said master token ~~is~~ to be used by the software task to initiate the ~~proper~~ unlock handshake to unlock the partition.

6. (Canceled)

7. (Currently Amended) The method of claim 1, wherein the software receives a usage token from ~~an~~ the IDE controller when the partition is unlocked and the ~~access~~ unlock handshake between the software and the IDE controller ~~is established by~~ includes passing the usage token back to the IDE controller as a parameter.

8. (Currently Amended) The method of claim 1, further comprising:

locking the partition in response to a lock request received from a software task having knowledge about a ~~proper~~ lock handshake for locking the partition.

9. (Currently Amended) The method of claim 1, further comprising:

providing a standard partition on the storage device, wherein said standard partition is always visible to the operating system and generally accessible to other software[[s]].

10. (Currently Amended) A machine-readable medium that provides instructions, which when executed by a set of processors, causes said set of processors to perform operations comprising:

receiving an open request from a software task to access a secure-private partition on an IDE hard drive of a computer system;

validating the open request received from the software task;

requesting unlocking of the secure-private partition in response to the validation of the open request received from the software task;

unlocking the secure-private partition in response to the unlocking request such that the partition that was previously invisible to an operating system becomes visible to the operating system; and

causing an IDE controller to prevent~~ing an~~ access to the secure-private partition when the secure-private partition is [[un]] locked unless the access is requested by a software task having knowledge about ~~a proper~~ an unlock ~~access~~ handshake for accessing the secure-private partition, the unlock handshake to include altering an

electrical signal on an interface line of the IDE controller, the electrical signal to cause the operating system to be granted access to the partition.

11. (Original) The machine-readable medium of claim 10, wherein the operations further comprise requesting locking of the secure-private partition in response to a close request received from the software.

12. (Original) The machine-readable medium of claim 10, wherein the requesting of the unlocking of the secure partition further comprises:

requesting a master token from an IDE controller when the computer system is turned on;

storing the master token in a secure storage location;

retrieving the master token from the secure storage location when an access to a secure-private partition is needed; and

passing the master token as a parameter to the IDE controller.

13. (Original) The machine-readable medium of claim 10, wherein the operations further comprise requesting an access to the secure-private partition in response to an access request received from the software.

14. (Currently Amended) The machine-readable medium of claim 13, wherein ~~the~~ requesting ~~of the~~ access to the secure partition further comprises:

receiving a usage token; and

passing the usage token to the IDE controller to gain an access to the secure partition.

15. (Original) The machine-readable medium of claim 10, wherein the request from the software to access the secure-private partition is received by a privacy gatekeeper which prescreens the request to determine if the software has an authorization to access the secure-private partition.

16-20. (Cancelled)

21. (Currently Amended) The method of claim 1, further comprising:

preventing an access to the partition when the partition is unlocked unless the access is requested by a software ~~having knowledge about a proper access~~ that performs another handshake for accessing the partition.

22. (Cancelled)

23. (Currently Amended) A method comprising:

partitioning a hard disk into a standard partition and a secure-private partition (SPP), the SPP operable in a locked mode and an unlocked mode;

switching the SPP from the locked mode to the unlocked mode in response to a handshake, the handshake including altering an electrical signal on an IDE controller interface line causing the partition to be unlocked;

receiving at least one read/write request from a requesting software program; and

switching the SPP from the unlocked mode to the locked mode in response to a close request; wherein

each of the at least one read/write requests is accompanied by a usage token.

24. (Currently Amended) The method of claim 23 wherein the handshake comprises:

receiving a secure token from a requesting software program;

verifying the secure token; and

returning ~~a~~ the usage token to the requesting software program.

25. (Previously Presented) The method of claim 23, further comprising:

validating the usage token received with a read/write request, and, if the token is valid, performing the request; or

if the token is invalid, denying the request.

26. (Previously Presented) The method of claim 23, further comprising:

generating a new usage token after the read/write request; and

returning the new usage token to the requesting software program.

27. (New) A system comprising:

a first token;

a hard disk drive including a disk controller and a secure-private partition;

a handshake connection to toggle the secure-private partition between visible mode and invisible mode;

an Integrated Drive Electronics (IDE) controller to initiate the handshake connection between the IDE controller and the disk controller upon receipt of the first token, the handshake connection including alteration of an electrical signal between the IDE controller and the disk controller.

28. (New) The system of claim 27 further comprising:

a requesting software task to request access to the secure-private partition.

29. (New) The system of claim 28 wherein an operating system is to launch the requesting software task.

30. (New) The system of claim 27 wherein the IDE controller is to validate the first token before initiating the handshake connection.

31. (New) The system of claim 30 wherein the IDE controller is to validate the first token by passing the first token through an internal hash mechanism.

32. (New) The system of claim 27 wherein the system includes handshake logic to strobe address and data lines high and to pass a second token through the data lines with the address lines pulled low.

33. (New) The system of claim 30 wherein the secure-private partition is to remain in invisible mode if the first token is invalid.